ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Апрель—июнь 2000. Серия 1. Том 7, № 2, 21–46

УДК 519.174

О СЛОЖНОСТИ НАХОЖДЕНИЯ СВЯЗНОЙ ПРЕДПИСАННОЙ РАСКРАСКИ ВЕРШИН ГРАФА*)

А. В. Кононов, С. В. Севастьянов

Рассматривается задача о связной раскраске вершин графа в предписанные цвета (СПР). Приводится подробный анализ сложности задачи в зависимости от значений четырех параметров: тип графа, максимальная степень графа, максимальная мощность предписания и максимальная частота вхождения цвета в предписания вершин. Для классов входов задачи СПР, определяемых значением четырехмерного характеристического вектора, находится система из восьми базисных классов, три из которых являются минимальными NP-полными, а остальные пять классов — максимальными полиномиально разрешимыми. Доказывается полнота представленной системы, т. е. сводимость любого класса входов к одному из восьми базисных классов. Показывается связь между задачей СПР и задачами дискретной оптимизации: задачей нахождения связных областей обслуживания, задачами построения расписаний на параллельных машинах, задачей нахождения максимального паросочетания в двудольном гиперграфе, задачей нахождения подсемейств векторов с суммой из заданной области.

Введение

В. Г. Визинг в работе [2] рассматривал следующую задачу о связной раскраске вершин графа в предписанные цвета (будем называть ее задачей СПР).

Пусть задан обыкновенный граф $G = (V, E)$ и конечное множество C, элементы которого называются цветами. Задана функция $A : V \to 2^C$, которую будем называть *вершинным предписанием* или просто *предписанием.* Предписание A каждой вершине $v \in V$ ставит в соответствие подмножество $A(v) \subseteq C$ *допустимых цветов*. Таким образом, каждый вход I задачи СПР задается тройкой (G, C, A). Функцию $\rho : V \to C$ будем называть *раскраской* вершин графа G, если $\rho(v) \in A(v)$ для каждой вершины $v \in V$. Раскраска называется *связной*, если каждый подграф, порожденный множеством всех вершин одного цвета, связен.

*) Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (коды проектов 99–01–00510 и 99–01–00581).

В задаче СПР для заданного входа I требуется найти связную предписанную раскраску (СПР) либо доказать, что ее не существует.

С задачей СПР тесно связана проблема нахождения достаточных условий существования СПР: для каких графов и при каких ограничениях на вершинные предписания существует связная раскраска. В [2] этот вопрос исследовался с точки зрения ограничений на *мощность предписания*. Величины $A_{\min} = \min_{v \in V} |A(v)|$ и $A_{\max} = \max_{v \in V} |A(v)|$ будем называть соответственно *минимальной* и *максимальной* мощностью предписания A. Пусть $\alpha(G)$ — наименьшее натуральное число k такое, что при любом вершинном предписании минимальной мощности не менее k существует связная раскраска вершин графа G.

Число $\alpha(G)$ является важной характеристикой графа G, когда исследуется возможность нахождения его связной предписанной раскраски. Приведем несколько простых примеров графов, для которых эта характеристика легко вычисляется.

Если G — полный граф, то $\alpha(G) = 1$, так как в полном графе любая раскраска является связной. Если G является n-вершинной цепью, то $\alpha(G) = \lceil \frac{n}{2} \rceil$. Наконец, максимальное значение числа $\alpha(G)$ среди всех связных n-вершинных графов G достигается на звезде и равно $n - 1$. (Последние два факта являются следствиями соотношения $\alpha(G) = \varepsilon(G)$, доказанного в теореме 3 из [2] для любого двудольного графа G, где $\varepsilon(G)$ — *неплотность* графа G, т. е. мощность наибольшего множества попарно несмежных вершин.) Очевидно также, что максимум $\alpha(G)$ среди всех n-вершинных графов равен n и достигается на графе G, состоящем из n изолированных вершин. Интересен вопрос вычисления (или оценки, если не удается найти точное значение) параметра $\alpha(G)$ для графов из конкретных классов. В [2] исследовались свойства $\alpha(G)$ и приводились его оценки в терминах других характеристик графа. Есть основания предполагать, что вычисление параметра $\alpha(G)$ для произвольного графа G является NP-трудной проблемой.

В настоящей работе мы коснемся другой алгоритмической проблемы: проверки существования связной раскраски для заданного графа G и заданного предписания A. Естественно было предполагать, что для произвольного графа G эта проблема окажется NP-полной. В. Г. Визинг высказал предположение, что она является NP-полной даже для простой цепи. Как будет видно из дальнейшего, это предположение оказалось верным. В данной работе мы проводим подробный сложностной анализ этой проблемы в зависимости от значений следующих четырех параметров: тип графа G, максимальная степень графа G, максимальная мощность предписания A и максимальная частота вхождения цвета в предписание A. Указанные четыре параметра

объединяются в характеристический вектор χ_I, зависящий от входа I. Для различных значений четырехмерного вектора x определяются классы входов $\mathscr{I}(x) = \{I \mid \chi_I \leqslant x\}$. Класс входов $\mathscr{I}$ будем называть *полиномиально разрешимым* (NP-*полным*), если задача СПР на этом классе входов является полиномиально разрешимой (NP-полной).

В разд. 1 проводится полная классификация классов $\mathscr{I}(x)$ по сложности для всех значений вектора x. Выделяется *базисная* система из восьми классов $\mathscr{I}(x_i)$ $(i = 1, \ldots, 8)$, пять из которых являются полиномиально разрешимыми, а три — NP-полными. (Список базисных классов приведен в таблице.) Доказывается *полнота* этой системы (т. е. объединение восьми областей, определяемых базисными классами, покрывает множество классов для всех допустимых значений характеристического вектора), ее *корректность* (непересекаемость областей NP-полных классов с областями полиномиально разрешимых классов) и *независимость* (т. е. никакой полиномиально разрешимый базисный класс не содержится в объединении областей, определяемых остальными четырьмя полиномиально разрешимыми базисными классами; таким же свойством обладают три NP-полных базисных класса).

Отсюда следует, во-первых, что каждый полиномиально разрешимый базисный класс является **максимальным**, т. е. расширение класса за счет увеличения любого из четырех параметров приводит к NP-полному классу; во-вторых, каждый NP-полный базисный класс является **минимальным** (сужение класса за счет уменьшения любого из четырех параметров приводит к полиномиально разрешимой задаче); и, в-третьих, представленная система из восьми базисных классов является **минимально возможной** при классификации входов на основе выбранных четырех параметров.

Далее в разд. 2 показывается, что задача о связной предписанной раскраске является «близкой родственницей» некоторых задач из других областей дискретной математики, таких как задача о существовании расписания без простоев и задержек для системы из m параллельных машин, задача о существовании паросочетания в двудольном гиперграфе, покрывающего одну из долей, задача нахождения подсемейства заданного семейства $(0, 1)$-векторов с суммой из заданной выпуклой области, наконец, задача о размещении предприятий на сети с требованием связности областей обслуживания.

В разд. 3 приводятся доказательства основных результатов для восьми базисных случаев (полиномиальная разрешимость и NP-полнота).

Наконец, в разд. 4 формулируются выводы и открытые вопросы.

1. Система базисных классов

Напомним, что степенью вершины v в графе G называется число инцидентных ей ребер. Максимальную степень вершины в графе G будем обозначать через $\Delta(G)$. Множество вершин $\mathscr{O}(i) = \{v \in V \mid i \in A(v)\}$ будем называть *областью вхождения* цвета i в предписании A, величину $f(i) = |\mathscr{O}(i)|$ — *частотой вхождения цвета* i в подмножества предписания A, а величину $f(A) = \max\limits_{i \in C} f(i)$ — *максимальной частотой* вхождения цвета в подмножества предписания A. Произвольные подмножества вершин в графе будем также называть *областями*. Будем говорить, что область $\mathscr{O}'$ связна, если порожденный ею подграф $G(\mathscr{O}')$ связен. Область $\mathscr{O}''$ будем называть *связной компонентой* области $\mathscr{O}'$, если подграф $G(\mathscr{O}'')$ является связной компонентой графа $G(\mathscr{O}')$. Для графа G определим его *тип* $T(G)$:

$$T(G) = \begin{cases} 0, & \text{если } G \text{ — лес;} \\ 1 & \text{в противном случае.} \end{cases}$$

Для каждого входа $I = (G, C, A)$ задачи СПР определим его четырехмерный *характеристический вектор* $\chi_I = (\chi_I(1), \chi_I(2), \chi_I(3), \chi_I(4))$: $\chi_I(1) = T(G)$, $\chi_I(2) = \Delta(G)$; $\chi_I(3) = A_{\max}$; $\chi_I(4) = f(A)$. Обозначим $N_k = \{k, k+1, \dots, \infty\}$. Тогда множество $X \doteq \{0,1\} \times N_0 \times N_1 \times N_1$ содержит все возможные значения характеристического вектора χ_I.

Для двух векторов x', x'' будем писать $x' \leqslant x''$, если $x'(i) \leqslant x''(i)$ для каждого $i = 1, \dots, 4$. Для заданного вектора $x' \in X$ через $\mathscr{I}(x')$ будем обозначать класс входов I с характеристическими векторами $\chi_I \leqslant x'$. Ясно, что из $x' \leqslant x''$ вытекает $\mathscr{I}(x') \subseteq \mathscr{I}(x'')$. Отсюда следует, что если установлена полиномиальная разрешимость класса входов $\mathscr{I}(x'')$, то из нее вытекает полиномиальная разрешимость класса $\mathscr{I}(x')$. Наоборот, из NP-полноты класса $\mathscr{I}(x')$ следует NP-полнота класса $\mathscr{I}(x'')$.

Для каждого вектора $x' \in X$ определим области $D^-(x') = \{x \in X \mid x \leqslant x'\}$ и $D^+(x') = \{x \in X \mid x \geqslant x'\}$. Тогда если класс $\mathscr{I}(x')$ полиномиально разрешим, то $\mathscr{I}(x)$ полиномиально разрешим для любого вектора $x \in D^-(x')$, а если $\mathscr{I}(x')$ является NP-полным, то $\mathscr{I}(x)$ также NP-полный класс при любом $x \in D^+(x')$.

В таблице приведены восемь различных векторов x_i $(i = 1, \dots, 8)$, определяющих базисную систему классов входов задачи СПР. Векторы $\{x_i \mid i = 1, \dots, 5\}$ определяют пять областей $D^-(x_i)$, объединение которых содержит множество всех таких векторов $x \in X$, что класс $\mathscr{I}(x)$ является полиномиально разрешимым. Векторы $\{x_i \mid i = 6, 7, 8\}$ определяют области $D^+(x_i)$, $i = 6, 7, 8$, объединение которых содержит множество всех векторов $x \in X$ таких, что класс $\mathscr{I}(x)$ является NP-полным. (Такое свойство семейства классов будем называть *полнотой*.)

Базисная система классов входов задачи СПР

Класс $\mathscr{I}(x_i)$, $x_i = \ldots$	Тип графа	$\Delta(G)$	$A_{\max}$	$f(A)$	Сложность
x_1	1	0	∞	∞	P
x_2	1	∞	∞	2	P
x_3	1	∞	1	∞	P
x_4	1	2	2	∞	P
x_5	1	3	2	3	P
x_6	0	1	3	3	NPC
x_7	0	4	2	3	NPC
x_8	0	3	2	4	NPC

Докажем, что приведенная в таблице система из восьми классов входов образует *базис*, т. е. обладает свойствами полноты, корректности и независимости.

Для начала докажем ее полноту, т. е.

$$X = \left(\bigcup_{i=1}^{5} D^-(x_i)\right) \bigcup \left(\bigcup_{i=6}^{8} D^+(x_i)\right).$$

Нетрудно видеть, что

$$\begin{gathered}
X \setminus \left(\bigcup_{i=1}^{3} D^-(x_i)\right) \subseteq \{x \in X \mid x(2) \geqslant 1,\ x(3) \geqslant 2,\ x(4) \geqslant 3\} \doteq X_1; \\
X_1 \setminus D^+(x_6) \subseteq \{x \in X \mid x(2) \geqslant 1,\ x(3) = 2,\ x(4) \geqslant 3\} \doteq X_2; \\
X_2 \setminus D^-(x_4) \subseteq \{x \in X \mid x(2) \geqslant 3,\ x(3) = 2,\ x(4) \geqslant 3\} \doteq X_3; \\
X_3 \setminus D^+(x_8) \subseteq \{x \in X \mid x(2) \geqslant 3,\ x(3) = 2,\ x(4) = 3\} \doteq X_4; \\
X_4 \setminus D^+(x_7) \subseteq \{x \in X \mid x(2) = 3,\ x(3) = 2,\ x(4) = 3\} \subset D^-(x_5).
\end{gathered}$$

Для доказательства корректности системы (т. е. того, что никакой класс не является одновременно полиномиально разрешимым и NP-полным) достаточно убедиться, что никакая область $D^-(x_i)$, $i = 1, \ldots, 5$, не пересекается ни с одной из областей $D^+(x_j)$, $j = 6, 7, 8$. Действительно,

$$D^-(x_5) \subseteq \{x \in X \mid x(3) \leqslant 2\},\ D^+(x_6) \subseteq \{x \in X \mid x(3) \geqslant 3\},$$

откуда следует, что $D^-(x_5) \cap D^+(x_6) = \varnothing$. Непересекаемость остальных пар областей проверяется непосредственно.

Наконец, для доказательства независимости системы достаточно проверить, что ни один вектор x_i, $i = 1, \ldots, 5$, не содержится ни в одной из областей $D^-(x_j)$, $j \in \{1, \ldots, 5\} \setminus \{i\}$, и ни один вектор x_i, $i = 6, 7, 8$, не содержится ни в одной из областей $D^+(x_j)$, $j \in \{6, 7, 8\} \setminus \{i\}$. Проверка выполняется непосредственно.

2. Близкие постановки задач

а. Задача о связных областях обслуживания

Задача СПР допускает следующую естественную интерпретацию. Множество вершин графа G представляет собой множество пунктов потребления (какого-то ресурса, услуг и т. п.). Для каждого пункта потребления v выбирается единственный поставщик из списка $A(v)$ допустимых для этого пункта поставщиков. При этом часто естественным требованием в задаче является требование связности области обслуживания для каждого поставщика.

Близкой к этой постановке является *простейшая задача размещения* (ПЗР), в которой каждому пункту потребления v требуется назначить поставщика $\rho(v) \in C$. Вместо предписанного множества поставщиков $A(v)$ задается предпочтение на множестве поставщиков в виде стоимости $g_{i,v}$ обслуживания пункта v поставщиком i. Суммарная стоимость обслуживания $\sum\limits_{v \in V} g_{\rho(v),v}$ входит слагаемым в целевую функцию $\Phi(\rho)$, определенную согласно (1). Кроме стоимости обслуживания в целевую функцию $\Phi(\rho)$ входит также стоимость $g_i(\rho)$ «использования» каждого поставщика $i \in C$, которая равна g_i^0, если $\rho^{-1}(i) \neq \varnothing$, и равна 0 в противном случае.

Нетрудно видеть, что для любого входа $I = (G, C, A)$ задачи СПР может быть определен вход ПЗР, который обладает следующим свойством: конечные значения целевой функции $\Phi(\rho)$ в ПЗР достигаются на тех и только тех назначениях ρ, которые допустимы относительно предписания A в задаче СПР. Действительно, достаточно определить матрицу $(g_{i,v})$ с коэффициентами из $\{0, \infty\}$ так, что

$$g_{i,v} = \begin{cases} 0, & i \in A(v), \\ \infty, & i \notin A(v). \end{cases}$$

Хотя требование связности областей обслуживания не входит напрямую в число требований, предъявляемых к решению ПЗР, удается показать [3], что если матрица $(g_{i,v})$ удовлетворяет определенным условиям, то всегда существует оптимальное решение ПЗР, в котором все области обслуживания связны (такое решение называем *связным*). В связи с этим А. А. Агеевым [5] был определен класс $\mathrm{Con}(G, C)$ матриц $(g_{i,v})$

($i \in C$, $v \in V$) таких, что при любом векторе $(g_1^0, \ldots, g_{|V|}^0)$ существует оптимальное связное решение задачи минимизации функции

$$\Phi(\rho) = \sum_{i\in C} g_i(\rho) + \sum_{v\in V} g_{\rho(v),v}. \tag{1}$$

В [5] наряду с задачей минимизации функции $\Phi(\rho)$ рассматривалась задача минимизации функции

$$\Omega(\rho) = \sum_{i\in C} g_i^0 h(\rho^{-1}(i)) + \sum_{v\in V} g_{\rho(v),v}, \tag{2}$$

где $h(\rho^{-1}(i))$ — число компонент связности подграфа, порожденного подмножеством вершин $\rho^{-1}(i)$. Тем самым в функцию $\Omega(\rho)$ как бы заложен штраф за несвязность обслуживания. В случае, когда матрица $(g_{i,v})$ принадлежит $\mathrm{Con}(G, C)$, решение задачи минимизации функции (2) является связным решением задачи ПЗР. Однако следует заметить, что в интересующем нас случае $(0, \infty)$-матриц $(g_{i,v})$ минимизация функции $\Omega(\rho)$ может не привести к нахождению назначения, которое было бы допустимо не только относительно предписания A, но и относительно требования связности (даже если такое назначение существует). Дело в том, что в задаче минимизации функции (2) мы заинтересованы в минимизации **суммарного** (по всем $i \in C$) взвешенного числа компонент связности, в то время как в задаче СПР требуется минимизировать **максимальное** (по $i \in C$) число компонент связности в областях $\rho^{-1}(i)$. Отсюда ясно, что если вместо функции $\Omega(\rho)$ ввести похожую функцию

$$\Omega'(\rho) = \max_{i\in C} h(\rho^{-1}(i)) + \sum_{v\in V} g_{\rho(v),v},$$

то решение задачи ее минимизации даст ответ на вопрос о существовании связной предписанной раскраски в задаче СПР, а именно: для заданной пары (G, A) связная предписанная раскраска существует, если и только если существует назначение ρ такое, что $\Omega'(\rho) = 1$ (матрица $(g_{i,v})$ со значениями из $\{0, \infty\}$ определяется по предписанию A). Таким образом, доказываемые в разд. 3 результаты о сложности задачи СПР распространяются на задачу минимизации функции $\Omega'(\rho)$.

б. Расписание на параллельных машинах без простоев

Для задачи СПР на семействе цепей (эта задача, как будет показано в разд. 3, является NP-полной) существуют другие эквивалентные постановки, в частности формулируемая ниже задача ПМ из теории расписаний.

Имеется m параллельных машин и одна работа, состоящая из n последовательных операций единичной длины. Каждая операция может

выполняться на любой машине. Для каждой машины задано семейство временных интервалов, когда она доступна. Вопрос: существует ли расписание выполнения работы на m машинах, при котором каждая машина работает без простоев (т. е. занята в течение связного интервала времени)?

Покажем, что задача СПР на семействе цепей полиномиально сводится к задаче ПМ. Сначала от задачи на семействе цепей перейдем к эквивалентной задаче на связной цепи. Пусть исходный граф $G = (V, E)$ является совокупностью l цепей. Соединим их в единую цепь $G' = (V', E')$, добавив $l-1$ промежуточных вершин $\{w_1, \ldots, w_{l-1}\}$ к множеству V и соединив вершину w_j $(j = 1, \ldots, l-1)$ с конечной вершиной j-й цепи и начальной вершиной $(j+1)$-й цепи. В предписание вершины w_j $(j = 1, \ldots, l-1)$ включим единственный цвет u_j, причем такой, что он не входит в предписания других вершин. Таким образом, новое множество цветов C' состоит из $m' \doteq m+l-1$ цветов: $C' = C \cup \{u_1, \ldots, u_{l-1}\}$.

Определение. Две вершины $v', v'' \in \mathscr{O}(i)$ будем называть *связными по цвету* i, если v' и v'' принадлежат одной компоненте связности области $\mathscr{O}(i)$.

Нетрудно видеть, что для любого цвета $i \in C'$ две различные вершины $v', v'' \in G'$ связны по цвету i, если и только если они являются вершинами графа G и связны в нем по цвету i. Таким образом, для графа G' существует СПР, если и только если существует СПР для графа G.

Занумеруем вершины цепи G' слева направо числами от 1 до $n' = n + l - 1$. Отметим, что подмножество вершин $\widetilde{V} \subseteq V'$ образует связную область, если и только если их номера образуют интервал целых чисел. Каждому цвету $i \in C'$ поставим в соответствие машину M_i. Машины M_i, $i \in C'$, недоступны вне интервала времени $[0, n']$, причем машина M_i доступна в k-м единичном интервале $[k-1, k]$ если и только если $i \in A(v_k)$. Имеется также работа, состоящая из n' операций единичной длины. Таким образом, эту работу требуется выполнить в интервале времени $[0, n']$. При этом каждая машина M_i должна работать без простоев.

Очевидно, что допустимое расписание в задаче ПМ существует, если и только если в исходной задаче СПР существует связная предписанная раскраска.

Заметим, что допустимое расписание оказывается также расписанием без задержек (т. е. вся работа выполняется за время, равное ее длительности). Поэтому задача СПР сводится также к задаче ПМ$'$, в которой допустимое расписание не должно содержать ни простоев машин, ни задержек работ. В теории расписаний известны постановки

задач с такими требованиями на «обоюдную беспростойность» расписания. В частности, в [1] рассматривалась задача open shop с длинами операций из множества $\{0,1\}$ и требованием обоюдной беспростойности. Было показано, что эта задача формулируется в терминах правильной интервальной раскраски ребер двудольного графа.*) В [4] показано, что задача нахождения интервальной раскраски является NP-полной проблемой в классе всех двудольных графов. В [6] приведена реальная (взятая из практики) интерпретация этой задачи в виде нахождения обоюдно беспростойного расписания консультаций родителей у преподавателей колледжа.

в. Максимальное паросочетание в двудольном гиперграфе

Задачу СПР на произвольном графе $G = (V, E)$ сведем к задаче о нахождении максимального паросочетания в двудольном гиперграфе. По заданному входу задачи СПР построим двудольный гиперграф H следующим образом. Вершины его левой доли образованы из вершин графа G. Вершинами правой доли являются элементы множества C: $i_1, \ldots, i_m$. Ребрами гиперграфа H являются все подмножества вершин вида $\{V'; i_s\}$ такие, что $V' \subseteq V$; $i_s \in C$, $i_s \in A(v)$ для любой вершины $v \in V'$; подграф графа G, порожденный подмножеством вершин V', связен. *Паросочетанием* в гиперграфе называется семейство гиперребер, непересекающихся по вершинам. Требуется найти паросочетание в гиперграфе H, покрывающее его левую долю.

Нетрудно видеть, что связная раскраска в задаче СПР существует, если и только если в гиперграфе H существует паросочетание, покрывающее его левую долю.

Заметим, что сведение задачи СПР к задаче о нахождении паросочетания в двудольном гиперграфе не полиномиально, так как длина записи второй задачи может быть экспонентой от длины записи исходной задачи (за счет большого количества гиперребер). Тем не менее в каждом из трех базовых случаев, в которых будет доказана NP-полнота задачи СПР, это сведение полиномиально, что позволяет переносить результаты об NP-полноте на задачу об отыскании паросочетания.

*) Напомним, что раскраска вершин (ребер) называется *правильной*, если смежные вершины (ребра) получают различные цвета. В этом смысле определенная выше связная раскраска вершин является «неправильной». Раскраска ребер называется *интервальной*, если множество цветов, используемых для раскраски ребер, инцидентных произвольной вершине, является интервалом целых чисел.

г. Подсемейство $(0,1)$-векторов с суммой из заданной области

Сформулируем задачу о нахождении *подсемейства векторов с суммой из заданной области* (ПВСЗО). Имеется семейство векторов $X = \{x_1, \dots, x_n\}$, принадлежащих пространству $\mathbb{R}^m$, и задана выпуклая область $D \subset \mathbb{R}^m$. Требуется определить, существует ли подсемейство $X' \subseteq X$ такое, что $\sum\limits_{x_j \in X'} x_j \in D$.

Переформулируем задачу о паросочетании в гиперграфе в терминах задачи ПВСЗО. Каждому гиперребру $e \in H$ поставим в соответствие вектор x_e с компонентами $x_e(v)$, $v \in V \cup C$:

$$x_e(v) = \begin{cases} 1, & \text{если } v \in e, \\ 0 & \text{в противном случае.} \end{cases}$$

Таким образом,

$$X = \{x_e \mid e \in H\};$$
$$D = \{x \in \mathbb{R}^m \mid x(v) = 1,\ \forall v \in V;\ x(v) \leqslant 1,\ \forall v \in C\}; \quad m = |V \cup C|.$$

Нетрудно видеть, что существует взаимно однозначное соответствие между паросочетаниями в гиперграфе H, покрывающими левую долю, и подсемействами векторов X' с суммой из области D.

Перейдем к доказательству результатов, приведенных в таблице.

3. Восемь базисных классов входов задачи СПР

Сначала докажем NP-полноту классов $\mathscr{I}(x_i)$, $i = 6, 7, 8$. В каждом из трех случаев к задаче СПР будет сведена известная NP-полная задача **3-ВЫПОЛНИМОСТЬ**, которая формулируется следующим образом.

ВХОД: Семейство логических переменных $z_1, \dots, z_n$ и семейство элементарных дизъюнкций $C_1, \dots, C_m$, каждая из которых состоит не более чем из трех *литералов*, т. е. переменных или их отрицаний.

ВОПРОС: Существуют ли значения переменных, при которых все $C_1, \dots, C_m$ истинны?

Теорема 1. *Задача* СПР *на классе входов* $\mathscr{I}(x_6)$, *где* $x_6 = (0, 1, 3, 3)$, *является* NP-*полной.*

Доказательство. Пусть на входе задачи **3-ВЫПОЛНИМОСТЬ** задано семейство элементарных дизъюнкций $C_1, \dots, C_m$, где $C_i = (l_{i1} \vee l_{i2} \vee l_{i3})$, а l_{ij} — литералы. Без ограничения общности можем считать, что каждая элементарная дизъюнкция состоит из трех литералов. Вместо семейства элементарных дизъюнкций будем также говорить

о логической формуле $F = C_1 \& C_2 \& \cdots \& C_m$, которая истинна, если и только если все элементарные дизъюнкции заданного семейства истинны.

Занумеруем произвольным образом все вхождения каждой переменной (с отрицанием или без него) в формулу F, и пусть k_s — число вхождений переменной z_s в F. Заметим, что между литералами l_{ij} и вхождениями переменных z_s в C_i существует взаимно однозначное соответствие: по номеру переменной z_s $(s = 1, \ldots, n)$ и номеру ее вхождения $(\nu = 1, \ldots, k_s)$ однозначно восстанавливаются индексы соответствующего литерала l_{ij} (будем обозначать их через $i(s,\nu)$ и $j(s,\nu)$). Наоборот, по индексам i, j литерала l_{ij} однозначно восстанавливаются номер переменной и номер ее вхождения в формулу (будем обозначать их через $s(i,j)$ и $\nu(i,j)$).

По заданному входу задачи **3-ВЫПОЛНИМОСТЬ** построим вход задачи СПР. Каждому литералу l_{ij} поставим в соответствие цвет α_{ij} (все α_{ij} различны), а каждому $C_i = (l_{i1} \vee l_{i2} \vee l_{i3})$ — изолированную вершину w_i с множеством предписанных цветов $A(w_i) = \{\alpha_{i1}, \alpha_{i2}, \alpha_{i3}\}$.

Далее для каждой переменной z_s $(s = 1, \ldots, n)$, имеющей k_s вхождений в формулу F, определим k_s цепей $(v'_{s\nu}, v''_{s\nu})$, $\nu = 1, \ldots, k_s$, и k_s изолированных вершин $d_{s1}, \ldots, d_{sk_s}$. Таким образом, множество вершин полученного графа G является объединением множеств $\{w_1, \ldots, w_m\}$ и $\{v'_{s\nu}, v''_{s\nu}, d_{s\nu} \mid \nu = 1, \ldots, k_s;\ s = 1, \ldots, n\}$. Кроме того, каждой переменной z_s поставим в соответствие $2k_s$ цветов β^s_k $(k = 1, \ldots, 2k_s)$. Таким образом, множество цветов C является объединением множеств $\{\alpha_{ij}\}$ и $\{\beta^s_k\}$.

Вершине $v'_{s\nu}$ предписаны два цвета: $A(v'_{s\nu}) \doteq \{\beta^s_{2\nu-1}, \beta^s_{2\nu}\}$.

Вершине $d_{s\nu}$ предписаны цвета $\beta^s_{2\nu}$ и $\beta^s_{2\nu+1}$, где $\beta^s_{2k_s+1} \doteq \beta^s_1$.

Наконец, вершине $v''_{s\nu}$ также предписаны два цвета. Один из них (цвет $\alpha(v''_{s\nu})$) совпадает с цветом $\alpha_{i(s,\nu),j(s,\nu)}$. Другой цвет из $A(v''_{s\nu})$ (обозначим его через $\beta(v''_{s\nu})$) совпадает с одним из цветов множества $A(v'_{s\nu})$ и определяется по следующему правилу:

$$\beta(v''_{s\nu}) = \begin{cases} \beta^s_{2\nu-1}, & \text{если } l_{i(s,\nu),j(s,\nu)} = z_s; \\ \beta^s_{2\nu}, & \text{если } l_{i(s,\nu),j(s,\nu)} = \bar{z}_s. \end{cases}$$

Таким образом, вход задачи СПР определен. Нетрудно видеть, что так определенный вход $I = (G, C, A)$ принадлежит классу $\mathscr{I}(x_6)$, поскольку каждая компонента связности графа G является либо изолированной вершиной, либо ребром, а предписание A удовлетворяет соотношениям $|A(v)| \leqslant 3$ (при любом $v \in V$) и $|\mathscr{O}(i)| \leqslant 3$ (для любого $i \in C$). Покажем, что для входа I существует связная предписанная раскраска, если и только если существуют значения переменных, при которых формула F истинна.

Пусть для I существует СПР. Рассмотрим вершины, ассоциированные с переменной z_s. Заметим, что если вершина v'_{s1} окрашена в цвет β_2^s, то для вершины d_{s1} уже нельзя использовать этот цвет ввиду принадлежности вершин d_{s1} и v'_{s1} различным компонентам связности графа G. Следовательно, вершина d_{s1} окрашена в цвет β_3^s. Но тогда вершина v'_{s2} не может быть окрашена в цвет β_3^s и должна быть окрашена в цвет β_4^s и т. д. Получаем, что каждая вершина $v'_{s\nu}$, $\nu = 1, \ldots, k_s$, окрашена в свой «четный» цвет $\beta_{2\nu}^s$. Если же v'_{s1} окрашена в цвет β_1^s, то вершина d_{sk_s} окрашена в цвет $\beta_{2k_s}^s$, вершина v'_{sk_s} — в цвет $\beta_{2k_s-1}^s$, и т. д.; таким образом, каждая вершина $v'_{s\nu}$, $\nu = 1, \ldots, k_s$, окрашена в свой «нечетный» цвет $\{\beta_{2\nu-1}^s\}$. Следовательно, всякая допустимая СПР обеспечивает «согласованность» раскраски вершин $v'_{s\nu}$ $(\nu = 1, \ldots, k_s)$ в определенном выше смысле. Определим значение каждой логической переменной z_s:

$$z_s = \text{«каждая вершина } v'_{s\nu} \ (\nu = 1, \ldots, k_s) \text{ окрашена в свой нечетный цвет»} . \quad (3)$$

Рассмотрим вершину w_i. Пусть она получила цвет α_{ij}, соответствующий литералу $l_{ij} = l_{i(s,\nu),j(s,\nu)}$. Так как вершина $v''_{s\nu}$ несвязна с вершиной w_i, то $v''_{s\nu}$ не может использовать цвет α_{ij} и, следовательно, должна быть окрашена в цвет $\beta(v''_{s\nu})$. Далее рассмотрим два случая.

Если $l_{ij} = z_s$, то $\beta(v''_{s\nu}) = \beta_{2\nu-1}^s$. Следовательно, вершина $d_{s,\nu-1}$ (где под $d_{s,0}$ понимается d_{s,k_s}) должна быть окрашена в цвет $\beta_{2\nu-2}^s$ (соответственно цвет $\beta_{2k_s}^s$), отличный от $\beta_{2\nu-1}^s$, а все вершины $v'_{s\nu}$ $(\nu = 1, \ldots, k_s)$ — в свои нечетные цвета. Отсюда и из (3) следует, что

$$z_s = \mathbf{true}, \quad l_{ij} = \mathbf{true}.$$

Пусть теперь $l_{ij} = \bar{z}_s$. Тогда $v''_{s\nu}$ окрашена в цвет $\beta(v''_{s\nu}) = \beta_{2\nu}^s$, каждая вершина $d_{s,\nu}$ $(\nu = 1, \ldots, k_s)$ — в нечетный цвет, вершины $v'_{s\nu}$ $(\nu = 1, \ldots, k_s)$ — в четные цвета. Поэтому

$$z_s = \mathbf{false}, \quad l_{ij} = \mathbf{true}.$$

В обоих случаях литерал l_{ij} принимает значение **true**, что обеспечивает истинность элементарной дизъюнкции C_i. Таким образом, из существования СПР следует истинность всех C_i.

Докажем обратное утверждение. Пусть $\tilde{z}_s$ — значения переменных z_s $(s = 1, \ldots, n)$, при которых все C_i истинны. Покажем, что существует СПР графа G.

Для каждой переменной z_s со значением $z_s = \mathbf{true}$ все вершины $v'_{s\nu}$ окрасим в свои нечетные цвета, а вершины $d_{s\nu}$ — в соответствующие четные. Для переменных z_s со значением **false** поступим наоборот.

Каждую вершину $v''_{s\nu}$ красим в ее β-цвет, если это возможно (т. е. если этот цвет не использован для вершины $d_{s,\nu-1}$ или $d_{s,\nu}$); в противном случае вершину $v''_{s\nu}$ красим в цвет $\alpha(v''_{s\nu})$.

Остается убедиться, что для каждой вершины w_i, $i = 1, \dots, m$, хотя бы один цвет из ее предписания $\{\alpha_{i1}, \alpha_{i2}, \alpha_{i3}\}$ остается неиспользованным после покраски вершин $\{v'_{s\nu}, v''_{s\nu}, d_{s\nu} \mid \forall s, \nu\}$, а следовательно, этот цвет может быть использован для покраски вершины w_i. Действительно, из истинности элементарной дизъюнкции C_i следует, что хотя бы один из литералов l_{i1}, l_{i2}, l_{i3} принимает истинное значение. Пусть $l_{ij} = \mathbf{true}$. Рассмотрим два случая, когда это равенство возможно.

Случай 1: $(l_{i,j} = z_{s(i,j)})\&(z_{s(i,j)} = \mathbf{true})$. Так как все вершины $d_{s,k}$ $(k = 1, \dots, k_s)$ при $z_{s(i,j)} = \mathbf{true}$ получают «четные» цвета, то вершина $v''_{s\nu}$ могла быть окрашена (и следовательно, окрашена) в свой («нечетный») β-цвет $\beta(v''_{s(i,j),\nu(i,j)}) = \beta^s_{2\nu-1}$. Следовательно, цвет $\alpha(v''_{s\nu}) = \alpha_{ij}$ остался неиспользованным.

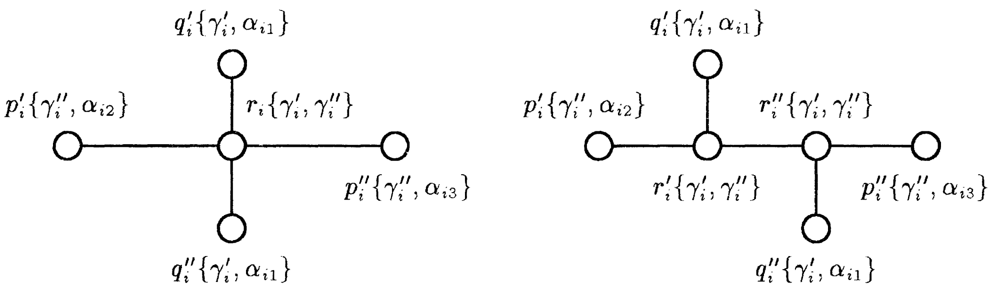
Случай 2: $(l_{i,j} = \bar{z}_{s(i,j)})\&(z_{s(i,j)} = \mathbf{false})$. Все вершины $d_{s,k}$ $(k = 1, \dots, k_s)$ красятся в «нечетные» цвета, но цвет $\beta(v''_{s(i,j),\nu(i,j)}) = \beta^s_{2\nu}$ — «четный». Следовательно, вершина $v''_{s\nu}$ окрашена в цвет $\beta(v''_{s\nu})$, а цвет $\alpha(v''_{s\nu}) = \alpha_{ij}$ остался неиспользованным.

Таким образом, все вершины графа G могут быть окрашены цветами из предписанных списков с соблюдением требования связности. Поскольку сведение задачи **3-ВЫПОЛНИМОСТЬ** к задаче СПР было полиномиально, то теорема 1 доказана.

Теорема 2. *Задача* СПР *на классе входов* $\mathscr{I}(x_7)$, *где* $x_7 = (0, 4, 2, 3)$, *является* NP-*полной.*

Доказательство. На этот раз задачу **3-ВЫПОЛНИМОСТЬ** произвольной формулы F сводим к задаче СПР со входом из класса $\mathscr{I}(x_7)$. Поскольку в этом классе разрешаются предписания мощности не более 2, то при построении входа задачи СПР вместо изолированных вершин w_i с предписаниями $A(w_i) = \{\alpha_{i1}, \alpha_{i2}, \alpha_{i3}\}$ мы используем изолированные подграфы G_i графа G, которые обладают аналогичным свойством, а именно: при условии существования допустимой раскраски в G обеспечивается занятость в подграфе G_i хотя бы одного из цветов $\alpha_{i1}, \alpha_{i2}, \alpha_{i3}$, соответствующих литералам l_{i1}, l_{i2}, l_{i3} элементарной дизъюнкции C_i. При этом в подграфе G_i допускается использование любого непустого подмножества цветов из $\{\alpha_{i1}, \alpha_{i2}, \alpha_{i3}\}$, что соответствует возможности любому непустому подмножеству литералов из C_i принимать значение **true**.

Подграф G_i изображен на рисунке (a), где при каждой вершине в фигурных скобках указаны предписанные ей цвета. Компоненты связности



а) $I \in \mathscr{I}(\boldsymbol{x}_7)$ б) $I \in \mathscr{I}(\boldsymbol{x}_8)$

Подграф G_i и его предписания

графа G, определяемые множеством вершин $\{v'_{s\nu}, v''_{s\nu}, d_{s\nu} \mid \forall\, s, \nu\}$, а также предписания этих вершин остаются без изменений.

Из приведенного рисунка видно, что в предписаниях вершин графа G появились новые цвета: в предписания вершин q_i', r_i и q_i'' входит цвет γ_i', а в предписания вершин p_i', r_i и p_i'' — цвет γ_i''. Цвета γ_i' и γ_i'' соответствуют элементарной дизъюнкции C_i и входят только в предписания вершин подграфа G_i. Нетрудно убедиться, что определенный таким образом вход I принадлежит классу $\mathscr{I}(x_7)$.

Пусть в графе G существует СПР. Если предположить, что в этой СПР ни один из цветов $\alpha_{i1}, \alpha_{i2}, \alpha_{i3}$ не используется для раскраски вершин подграфа G_i, то вершины q_i' и q_i'' получают цвет γ_i'. Следовательно, по требованию связности вершина r_i должна быть окрашена в тот же цвет γ_i'. С другой стороны, вершины p_i' и p_i'' получают цвет γ_i'' и, следовательно, вершина r_i должна быть окрашена в цвет γ_i'', что невозможно. Противоречие доказывает, что хотя бы один из цветов $\alpha_{i1}, \alpha_{i2}, \alpha_{i3}$ должен использоваться для раскраски вершин подграфа G_i. Также легко проверить, что в G_i возможно использование любого непустого набора цветов из $\{\alpha_{i1}, \alpha_{i2}, \alpha_{i3}\}$.

Дальнейшая схема доказательства теоремы 2 в основных чертах повторяет схему доказательства теоремы 1. Из наличия допустимой раскраски в G следует, что для каждой элементарной дизъюнкции C_i хотя бы одна из трех вершин $v''_{s(i,j,),\nu(i,j)}$ $(j = 1, 2, 3)$ должна быть окрашена в цвет, отличный от $\alpha(v''_{s\nu})$, т. е. в цвет $\beta(v''_{s\nu})$. Но из доказательства теоремы 1 следует, что вершина $v''_{s\nu}$ может быть окрашена в цвет $\beta(v''_{s\nu})$ только в следующих двух случаях:

— если $\beta^s_{2\nu-1} \in A(v''_{s\nu})$ и вершина $v'_{s\nu}$ окрашена в цвет $\beta^s_{2\nu-1}$, что соответствует литералу $l_{ij} = z_s$ и значению $z_s = \textbf{true}$ (переменные z_s вновь определяем согласно (3));

— если $\beta^s_{2\nu} \in A(v''_{s\nu})$ и вершина $v'_{s\nu}$ окрашена в цвет $\beta^s_{2\nu}$, что соответствует литералу $l_{ij} = \bar{z}_s$ и значению $z_s =$ **false**.

В обоих случаях литерал l_{ij}, а следовательно, и элементарная дизъюнкция C_i принимают истинное значение. Таким образом, при значениях переменных z_s, определенных согласно (3), формула F истинна.

Доказательство обратной импликации аналогично доказательству теоремы 1. Теорема 2 доказана.

Теорема 3. *Задача* СПР *на классе входов* $\mathscr{I}(x_8)$, *где* $x_8 = (0, 3, 2, 4)$, *является* NP-*полной.*

Доказательство. Для любого входа $I = (G, C, A) \in \mathscr{I}(x_8)$ в графе G запрещаются вершины степени больше 3 (подобные вершине r_i в подграфе G_i на рисунке (*а*)). Это ограничение нетрудно соблюсти, преобразовав граф G_i так, как показано на рисунке (*б*). При этом число вхождений цветов γ'_i, γ''_i в предписания вершин графа G_i увеличивается до 4. Поэтому вход I не принадлежит классу $\mathscr{I}(x_7)$, но принадлежит классу $\mathscr{I}(x_8)$. В остальных деталях доказательство теоремы 3 повторяет доказательство теоремы 2.

Перейдем к доказательству полиномиальной разрешимости задачи СПР на пяти оставшихся базисных классах входов.

Теорема 4. *Задача* СПР *на классе входов* $\mathscr{I}(x_1)$, *где* $x_1 = (1, 0, \infty, \infty)$, *полиномиально разрешима.*

Доказательство. Для любого входа $I \in \mathscr{I}(x_1)$ все вершины графа G принадлежат разным компонентам связности. Поэтому они должны быть окрашены в попарно различные цвета. Таким образом, задача СПР сводится к задаче об отыскании максимального паросочетания в двудольном графе $\widehat{G}$, левая доля которого состоит из вершин множества V, правая — из цветов множества C, а ребро $e = (i, j)$ соединяет вершину $i \in V$ с вершиной $j \in C$ если и только если $j \in A(i)$.

Нетрудно видеть, что СПР в графе G существует, если и только если максимальное паросочетание в G покрывает все вершины $i \in V$. Поскольку задача о нахождении максимального паросочетания в двудольном графе полиномиально разрешима [7], то задача СПР на классе входов $\mathscr{I}(x_1)$ полиномиально разрешима. Теорема 4 доказана.

Теорема 5. *Задача* СПР *на классе входов* $\mathscr{I}(x_2)$, *где* $x_2 = (1, \infty, \infty, 2)$, *полиномиально разрешима.*

Доказательство. Алгоритм состоит из двух этапов. На первом этапе последовательно просматриваются все цвета $i \in C$, и если область $\mathscr{O}_i$ связна, то все вершины $v \in \mathscr{O}_i$ красятся цветом i, после чего эти вершины удаляются из графа G и областей $\mathscr{O}_j$ для оставшихся цветов

j ($j \neq i$), а цвет i — из множества C. На втором этапе все оставшиеся в G вершины красятся в различные цвета так, как это делалось в доказательстве теоремы 4.

Покажем, что описанный алгоритм находит допустимую раскраску в задаче СПР, если таковая существует.

Пусть $\rho : V \to C$ — допустимая раскраска. Рассмотрим первый цвет и его область вхождения $\mathcal{O}_1$. Если $\mathcal{O}_1$ связна, то все вершины из $\mathcal{O}_1$ перекрасим в цвет 1. При этом область раскраски цвета 1 совпадает с $\mathcal{O}_1$ и является связной. Перекраска вершин в цвет 1 может, вообще говоря, нарушить связность областей раскраски других цветов. Однако в рассматриваемом случае (когда $I \in \mathscr{I}(x_2)$) область раскраски одним цветом состоит не более чем из двух вершин и остается связной после удаления из нее любого количества вершин. По этой же причине удаление вершин из области $\mathcal{O}_1$ в графе G не нарушает связности областей вхождения других цветов. Удаляем цвет 1 из C и переходим к цвету 2 и т. д. Очевидно, что после завершения первого этапа алгоритма область $\mathcal{O}_i$ каждого из оставшихся в C цветов является несвязной, т. е. состоит из двух несмежных вершин. Следовательно, в раскраске ρ эти вершины окрашены в разные цвета. Таким образом, для всех вершин, оставшихся в графе G после первого этапа, существует предписанная раскраска в различные цвета, которая находится алгоритмом второго этапа за полиномиальное время. Теорема 5 доказана.

Теорема 6. *Задача СПР на классе входов $\mathscr{I}(x_3)$, где $x_3 = (1, \infty, 1, \infty)$, полиномиально разрешима.*

Доказательство. Поскольку предписание $A(v)$ для каждой вершины v состоит из единственного цвета, то раскраска ρ, допустимая относительно предписания A, определяется однозначно. Таким образом, для установления связности раскраски достаточно для каждого цвета $i \in C$ проверить связность области $\mathcal{O}_i$ вхождения цвета $i \in C$ в предписания вершин, что может быть сделано за полиномиальное время. Теорема 6 доказана.

Для доказательства полиномиальной разрешимости следующих двух классов введем несколько новых понятий.

Маркировкой вершин графа G назовем функцию $\mu : V \to 2^C$, т. е. приписывание каждой вершине $v \in G$ некоторого подмножества цветов $\mu(v) \subseteq C$. Через $\mu^{-1}(i)$ будем обозначать *область маркировки цветом* i, т. е. $\mu^{-1}(i) = \{v \in V \mid i \in \mu(v)\}$. *Связной предписанной маркировкой* (СПМ) вершин графа $G = (V, E)$ назовем такую маркировку μ, что

- $\mu(v) \neq \varnothing,\ v \in V$;
- $\mu(v) \subseteq A(v),\ v \in V$;
- для любого цвета $i \in C$ область $\mu^{-1}(i)$ связна в G.

Маркировку μ, удовлетворяющую перечисленным выше требованиям, будем также называть *допустимой* относительно входа $I = (G, C, A)$.

Ясно, что любая СПР является частным случаем СПМ, когда вершинам приписываются одноэлементные множества цветов. Таким образом, из существования СПР для входа I следует существование СПМ для этого входа. Обратное, вообще говоря, не верно.

В задаче СПМ требуется построить алгоритм, который бы по заданному входу I либо находил СПМ, либо выдавал сообщение о том, что СПМ не существует.

Лемма 1. *Задача* СПМ *на классе входов* $I = (G, C, A)$ *с* $A_{\max} \leqslant 2$ *разрешима за время* $O(nu)$, *где* n *— число вершин, а* u *— число ребер графа* G.

Доказательство. Для решения задачи СПМ будет использован описываемый ниже алгоритм $\mathscr{A}_\mu(I)$, состоящий из этапа инициализации и цикла типа **repeat** ... **until false**. В алгоритме $\mathscr{A}_\mu(I)$ будут использоваться булевы переменные ПРОБМАР и УСПЕХ; ПРОБМАР $=$ **true** означает, что применяется пробная маркировка, допускающая ее последующую отмену; УСПЕХ $=$ **true** означает, что найдена СПМ.

Пусть $\mathscr{P}$ обозначает множество пар (V', A') таких, что $V' \subseteq V$; $A' : V \to 2^C$; $A'(v) \subseteq A(v)$, $v \in V$. Будем говорить, что $(V', A') \in \mathscr{P}$ меньше пары $(V'', A'') \in \mathscr{P}$, если $V' \subseteq V''$ и $A'(v) \subseteq A''(v)$ для любой вершины $v \in V$. Если при этом хотя бы одно из включений строгое, то (V', A') строго меньше (V'', A'').

Процесс маркировки вершин с помощью алгоритма $\mathscr{A}_\mu(I)$ будет сопровождаться изменением текущего множества немаркированных вершин V_T и текущего предписания A_T; при этом в каждый момент времени пара (V_T, A_T) будет принимать значения из $\mathscr{P}$.

Алгоритм $\mathscr{A}_\mu(I)$.

Инициализация

Задаем $(V_T, A_T) := (V, A)$; ПРОБМАР $:=$ **false**; УСПЕХ $:=$ **false**.

Из $A_{\max} \leqslant 2$ следует, что предписание $A(v)$ каждой вершины $v \in V$ состоит не более чем из двух цветов. Просматриваются все вершины из V_T и формируются списки V_0 и V_1, где V_i — список вершин $v \in V_T$ с $|A_T(v)| = i$. Задается исходная маркировка: $\mu_T(v) = \varnothing$ $(v \in V)$.

Цикл

На каждой итерации цикла проверяются следующие пять условий и выполняются соответствующие действия:

Условие 1. Если $V_T = \varnothing$, то $\{$УСПЕХ $:=$ **true**; **stop**$\}$ (найдена искомая СПМ).

Условие 2. Если $V_0 = \varnothing$ и $V_1 = \varnothing$, то выполнить

Действие 2 (начало пробной маркировки)

ПРОБМАР := **true**; $(\widetilde{V}, \widetilde{A}) := (V_T, A_T)$; $\tilde{\mu} := \mu_T$ — запоминается значение пары (V_T, A_T) и маркировки μ_T перед началом пробной маркировки; выбирается произвольная вершина $\tilde{v} \in V_T$, произвольный цвет $\tilde{i} \in A_T(\tilde{v})$ и применяется процедура ExpandColor$(\tilde{v}, \tilde{i})$.

Условие 3. Если $V_0 = \varnothing$ и $V_1 \neq \varnothing$, то выполнить

Действие 3

Выбирается произвольная вершина $v' \in V_1$ и для единственного цвета $i' \in A_T(v')$ применяется процедура ExpandColor(v', i').

Условие 4. Если $V_0 \neq \varnothing$ и ПРОБМАР = **true**, то выполнить

Действие 4 (отмена пробной маркировки)

ПРОБМАР := **false**; $(V_T, A_T) := (\widetilde{V}, \widetilde{A})$; $\mu_T := \tilde{\mu}$; $V_0 := \varnothing$; $V_1 := \varnothing$. Выбирается цвет $i'' \in A_T(\tilde{v})$, отличный от $\tilde{i}$, и применяется процедура ExpandColor$(\tilde{v}, i'')$.

Условие 5. Если $V_0 \neq \varnothing$ и ПРОБМАР = **false**, то {вывод: «Допустимой маркировки не существует»; **stop**}.

Конец цикла

Процедура ExpandColor(v', i').

Находится максимальная связная подобласть $\mathscr{O}'$ области $\mathscr{O}(i')$, содержащая вершину v'; все вершины из $\mathscr{O}'$ маркируются цветом i' и удаляются из V_T и V_1. Просматриваются оставшиеся вершины $v \in V_T$: если $i' \in A_T(v)$, то {цвет i' удаляется из $A_T(v)$, а вершина v помещается в список $V_{|A_T(v)|}$}.

Конец процедуры ExpandColor.

Покажем, что алгоритм $\mathscr{A}_\mu$ завершает работу за полиномиальное время. При этом если в графе G существует СПМ, то алгоритм $\mathscr{A}_\mu$ завершает работу со значением УСПЕХ = **true**, а полученная маркировка $\mu(v)$ (определенная для всех вершин $v \in V$, поскольку выполнено условие $V' = \varnothing$) является допустимой.

Работу алгоритма $\mathscr{A}_\mu$ до первой реализации действия 2, между двумя реализациями действия 2 (от начала одной реализации до начала следующей), а также после последней реализации действия 2 будем называть соответственно *начальным, промежуточным* и *конечным туром.* Вершину $\tilde{v}$ и цвет $\tilde{i}$, выбираемые при выполнении действия 2 текущего тура, будем называть *исходной вершиной* и *исходным цветом* тура.

Для доказательства конечности работы алгоритма прежде всего заметим, что на каждой итерации цикла выполнено одно из пяти условий. Выполнение условий 1 или 5 влечет остановку алгоритма. При выполнении одного из трех остальных условий осуществляются действия,

которые сопровождаются строгим уменьшением значения пары (V_T, A_T). Исключением является действие 4, в котором осуществляется возврат к исходному (для текущего тура) значению пары (V_T, A_T). Однако ясно, что между двумя реализациями действия 4 значение переменной ПРОБМАР должно измениться с **false** на **true**, а для этого необходимо выполнить действие 2. Из определения тура следует, что действие 4 выполняется в каждом туре не более одного раза. Поскольку непосредственно после действия 2 и непосредственно после действия 4 выполняется действие 3 (при выполнении которого множество V_T убывает хотя бы на одну вершину), то присвоение $(\widetilde{V}, \widetilde{A}) := (V_T, A_T)$ при каждой следующей реализации действия 2 порождает множество $\widetilde{V}$ с меньшим числом вершин. Следовательно, в алгоритме выполняется не более n туров.

Действие 3 может выполняться в туре многократно, но так как при этом осуществляется маркировка и удаление из множества V_T хотя бы одной вершины, то в каждом туре действие 3 выполняется не более $2n$ раз. С использованием списков инцидентных ребер, заданных для каждой вершины, и списков $\mathscr{O}(i)$ $(i \in C)$ можно организовать работу алгоритма $\mathscr{A}_\mu$ так, чтобы временная сложность каждого тура не превосходила $O(u)$, а общая сложность алгоритма $\mathscr{A}_\mu$ — $O(nu)$ операций, где u — число ребер графа G.

Теперь предположим, что для входа I существует допустимая маркировка μ. Докажем, что маркировка $\mu_{\mathscr{A}}$, найденная алгоритмом $\mathscr{A}_\mu$, является допустимой.

Допустимость $\mu_{\mathscr{A}}$ с точки зрения связности и соблюдения предписаний очевидна. Остается доказать, что окончательная маркировка $\mu_{\mathscr{A}}$ является полной. Для этого достаточно убедиться, что алгоритм завершит работу по условию 1 (когда $V_T = \varnothing$, т. е. все вершины маркированы).

Предположим противное, т. е. пусть в некотором туре (назовем его *конечным*) алгоритм завершит работу по условию 5. Покажем, что это предположение противоречит сделанному выше предположению о существовании СПМ для входа I.

Вход I будем называть *СПМ-положительным*, если для него существует СПМ.

Далее докажем три вспомогательных утверждения.

Утверждение 1. *Пусть A' и A'' — значения предписания A_T до и после выполнения процедуры* ExpandColor(v', i'). *Тогда если для входа $I' = (G, C, A')$ существует допустимая маркировка μ и цвет i' используется в маркировке $\mu(v')$ вершины v', то маркировка μ является допустимой для входа $I'' = (G, C, A'')$.*

Доказательство. Поскольку при выполнении процедуры мы распространяем маркер i' на максимальную связную область $\mathcal{O}' \subseteq \mathcal{O}(i')$, содержащую вершину v', то можем утверждать, что ни одна из вершин $v \notin \mathcal{O}'$ не маркирована в μ цветом i' (так как не существует связной области $\mathcal{O}'' \subseteq \mathcal{O}(i')$, включающей одновременно и вершину v', и вершину v). Следовательно, удаление цвета i' из предписания каждой такой вершины $v \notin \mathcal{O}'$ не нарушает допустимости маркировки μ, т. е. μ допустима относительно входа I''. Утверждение 1 доказано.

Утверждение 2. *Пусть (V', A') и (V'', A'') — значения пары (V_T, A_T) в начале и по окончании некоторого промежуточного тура. Тогда если вход $I' = (G, C, A')$ является* СПМ-*положительным, то вход $I'' = (G, C, A'')$ также* СПМ-*положителен.*

Доказательство. Пусть μ — СПМ входа I', а $\mu_{\mathscr{A}}$ — маркировка, полученная алгоритмом по окончании промежуточного тура. Убедимся, что мы можем допустимым образом доопределить маркировку $\mu_{\mathscr{A}}$ на множестве вершин V''. Действительно, так как по окончании промежуточного тура мы имеем $V_0 = \varnothing$, $V_1 = \varnothing$, то ни один из цветов $i \in A(v)$ в предписаниях вершин $v \in V''$ не был удален в процессе работы алгоритма $\mathscr{A}_\mu$. Это означает, что данные цвета еще не использовались в маркировке $\mu_{\mathscr{A}}$, а следовательно, никакой цвет $i \in A(v)$ $(v \in V'')$ не удалялся из предписаний остальных вершин $v \in V$. Таким образом, маркируя цветами $i \in A(v)$ $(v \in V'')$ те же множества вершин, что и в допустимой маркировке μ, мы допустимым образом доопределяем $\mu_{\mathscr{A}}$ на множестве вершин V''. Утверждение 2 доказано.

Утверждение 3. *Пусть в текущем туре алгоритма $\mathscr{A}_\mu$ выполняется действие 4 и A', A'' — значения A_T до начала и по окончании текущего тура. Тогда если вход $I' = (G, C, A')$ является* СПМ-*положительным, то вход $I'' = (G, C, A'')$ также* СПМ-*положителен.*

Доказательство. Пусть μ — допустимая маркировка для входа I', а $\tilde{v}$ и $\tilde{i}$ — вершина и цвет, выбранные исходными в текущем туре алгоритма $\mathscr{A}_\mu$. Предположим, что цвет $\tilde{i}$ используется в маркировке $\mu(\tilde{v})$. Тогда по утверждению 1 маркировка μ является допустимой и для входа $I_1 = (G, C, A_1)$, полученного по окончании действия 2. Если для входа I_1 имеем $V_1 \neq \varnothing$ и для последующего действия 3 выбрана вершина $v' \in V_1$, то очевидно, что единственный цвет $i' \in A_1(v')$ используется в маркировке $\mu(v')$, а следовательно, по утверждению 1, маркировка μ остается допустимой и для входа, полученного по окончании действия 3. Последующая серия действий 3 также с очевидностью приводит к СПМ-положительному входу и не может закончиться условием $V_0 = \varnothing$ (т. е. СПМ-отрицательным входом). Но это противоречит тому,

что в текущем туре выполняется действие 4. Значит, предположение о том, что цвет $\tilde{i}$ используется в допустимой маркировке $\mu(\tilde{v})$, было неверным. Но отсюда следует, что второй цвет из $A(\tilde{v})$ непременно используется в маркировке $\mu(\tilde{v})$ и, значит, как было показано выше, действие 4 и последующая серия действий 3 текущего тура заканчиваются СПМ-положительным входом. Утверждение 3 доказано.

Продолжим доказательство леммы 1. Поскольку по предположению вход I является СПМ-положительным, то согласно утверждению 1 серия действий 3, выполняемая в начальном туре, сохраняет СПМ-положительность текущего входа. В течение последующих промежуточных туров СПМ-положительность также сохраняется по утверждению 2. Наконец, так как мы предположили, что алгоритм завершает работу по условию 5, то отсюда следует, что в конечном туре выполняется действие 4. Следовательно, по утверждению 3 конечный тур также порождает СПМ-положительный вход. Но, с другой стороны, вход, получаемый по окончании работы алгоритма по условию 5, содержит вершину с пустым предписанием и не может иметь СПМ. Противоречие доказывает, что если первоначальный вход I имеет СПМ, то алгоритм $\mathscr{A}_\mu$ не может завершить работу по условию 5, а следовательно, он находит допустимую маркировку для входа I. Лемма 1 доказана.

Лемма 2. *Для любого входа $I \in \mathscr{I}(x_4)$ существует* СПМ *если и только если для входа I существует* СПР. *Существует алгоритм временной сложности $O(n)$, который по любому заданному входу $I \in \mathscr{I}(x_4)$ и его* СПМ *находит* СПР.

Доказательство. Для перестройки СПМ μ в СПР ρ достаточно устранить двойную маркировку некоторых вершин $v \in V$, оставив из найденной маркировки $\mu(v)$ каждой вершины лишь один цвет. Для входов $I \in \mathscr{I}(x_4)$ такая перестройка СПМ μ в СПР ρ может быть выполнена с помощью описываемого ниже алгоритма $\mathscr{A}^4_{\mu,\rho}(I)$.

Заметим, что для любого входа $I = (G, C, A) \in \mathscr{I}(x_4)$ степень каждой вершины в графе G не превосходит 2, поэтому каждая связная компонента графа G представляет собой простую цепь либо цикл. Таким образом, мы можем осуществлять последовательный обход вершин компоненты связности, начиная с любой заданной вершины, в одном из двух возможных направлений. Поскольку область $\mu^{-1}(i)$ вершин, маркированных цветом $i \in C$, связна, то для рассматриваемого класса входов она также является цепью либо циклом, а пересечение двух таких областей — для двух разных цветов i' и i'' — является циклом, цепью либо парой цепей. (Последний случай имеет место, когда каждая область $\mu^{-1}(i')$, $\mu^{-1}(i'')$ является цепью, содержащая их общая компонента связности является циклом и цепи $\mu^{-1}(i')$, $\mu^{-1}(i'')$ покрывают всю

компоненту связности, пересекаясь на своих концах.)

Алгоритм $\mathscr{A}^4_{\mu,\rho}(I)$

В алгоритме используются три различных процедуры просмотра вершин, которые будем называть *Просмотр 1, Просмотр 2 и Просмотр 3.* При этом Просмотр 1 является внешним циклом по вершинам графа, внутри которого вызываются процедуры Просмотр 2 и Просмотр 3.

Просмотр 1. Просматриваем вершины графа G в некотором порядке (например, по возрастанию их номеров). Если некоторая вершина $v' \in V$ имеет маркировку двумя цветами (i' и i''), то начинаем

Просмотр 2. Просматриваем вершины этой компоненты связности, отправляясь от вершины v' в одном из двух возможных направлений, пока не выполнится одно из двух условий:

1) встретилась вершина v'', не содержащая в своей маркировке хотя бы один из двух цветов $\{i', i''\}$;

2) встретилась вершина v'.

Пусть выполнилось условие 1 (для определенности считаем, что $i' \notin \mu(v'')$). Тогда начинаем

Просмотр 3. Просматриваем вершины компоненты связности, отправляясь от вершины v'' и двигаясь в направлении, обратном тому, который использовался в Просмотре 2. При этом устраняем цвет i' из маркировки просматриваемых вершин, пока не выполнится одно из двух условий:

1$'$) встретилась вершина v, не содержащая хотя бы один из двух цветов $\{i', i''\}$;

2$'$) встретился конец цепи (когда компонента связности является цепью).

Ясно, что одно из условий $1', 2'$ обязательно выполнится, поскольку если компонента связности является циклом (а не цепью), то условие $1'$ выполнится либо в вершине $v = v''$, либо раньше. При выполнении одного из условий $1', 2'$ прекращаем Просмотр 3 и продолжаем Просмотр 1.

Если Просмотр 2 закончился по условию 2, то компонента связности является циклом (C), все вершины которого маркированы обоими цветами i', i''. Для устранения двойной маркировки вершин $v \in C$ из маркировки всех вершин $v \in C$ достаточно убрать один и тот же (произвольный) цвет. После этого возвращаемся к Просмотру 1.

Алгоритм $\mathscr{A}^4_{\mu,\rho}(I)$ описан.

Пусть для входа $I \in \mathscr{I}(x_4)$ известна его допустимая маркировка μ. Для начала заметим, что описанный алгоритм $\mathscr{A}^4_{\mu,\rho}(I)$ имеет линейную

от n трудоемкость. Также ясно, что при удалении цвета i' из маркировки вершин $v \in V$ область $\mu^{-1}(i')$ остается связной. (Если эта область была циклом, то мы удаляем ее целиком, а если была цепью, то она сокращается на одном из концов этой цепи.) Поэтому полученная после удаления двойной маркировки раскраска является предписанной и связной. Тем самым доказано, что из существования СПМ вытекает существование СПР для заданного входа I. Если же СПМ не существует, то и СПР (как частного случая СПМ) также не существует. Лемма 2 доказана.

Лемма 3. *Для любого входа $I \in \mathscr{I}(x_5)$ существует* СПМ *если и только если для входа I существует* СПР. *Существует алгоритм временной сложности $O(n)$, который по любому заданному входу $I \in \mathscr{I}(x_5)$ и его* СПМ *находит* СПР.

Доказательство. Перестройка СПМ μ в СПР ρ для входов $I \in \mathscr{I}(x_5)$ может быть выполнена с помощью описываемого ниже алгоритма $\mathscr{A}^5_{\mu,\rho}(I)$.

Алгоритм $\mathscr{A}^5_{\mu,\rho}(I)$

Последовательно для каждой вершины $v' \in V$ делаем следующую проверку. Если $\mu(v')$ состоит из двух цветов (i' и i''), то проверяем маркировку вершин, смежных с v' (множество таких вершин обозначим через V'; из $I \in \mathscr{I}(x_5)$ имеем $|V'| \leqslant 3$). В результате проверки приходим к одному из следующих двух случаев:

— существует цвет $i \in \mu(v')$, который входит в маркировку не более чем одной вершины из V'; в этом случае удаляем цвет i из $\mu(v')$;

— каждый из цветов $i', i'' \in \mu(v')$ входит в маркировку двух вершин из V'; в этом случае найдется вершина $v'' \in V'$, которая маркирована обоими цветами i' и i''; удаляем цвет i' из $\mu(v')$ и $\mu(v'')$.

Алгоритм $\mathscr{A}^5_{\mu,\rho}(I)$ описан.

Для доказательства того, что для любого входа $I = (G, C, A) \in \mathscr{I}(x_5)$ алгоритм $\mathscr{A}^5_{\mu,\rho}(I)$ преобразует допустимую маркировку μ вершин графа G в допустимую раскраску ρ, достаточно убедиться, что удаление цвета из маркировки той или иной вершины не нарушает связности маркировки этим цветом. Действительно, для любого цвета $i \in C$ область $\mu^{-1}(i)$ состоит не более чем из трех вершин. Удаление какой-то вершины v из области $\mu^{-1}(i)$ (что происходит при удалении цвета i из маркировки вершины v) может нарушить связность этой области только в том случае, если

- $|\mu^{-1}(i)| = 3$;
- вершина v смежна с двумя другими вершинами из $\mu^{-1}(i)$.

(Вершину с указанными свойствами будем называть *средней* для цвета i.)

Если вершина v является средней лишь для одного из двух цветов маркировки $\mu(v)$, то удаление второго цвета из $\mu(v)$ не нарушает связности маркировки этим цветом. Таким образом, нарушение связности может возникнуть лишь тогда, когда вершина v является средней для обоих цветов из $\mu(v)$. Но тогда некоторая вершина v', смежная с v, маркирована теми же двумя цветами (т. е. $\mu(v') = \mu(v)$). Удаление цвета $i \in \mu(v)$ из маркировки обеих вершин v и v' не нарушает связности области $\mu^{-1}(i)$ по той простой причине, что после удаления из нее вершин v и v' в ней остается не более одной вершины.

Таким образом, раскраска ρ, получаемая по окончании работы алгоритма $\mathscr{A}^5_{\mu,\rho}(I)$, является СПР для входа I. Очевидно, что трудоемкость алгоритма оценивается величиной $O(n)$. Наконец, если для входа I не существует СПМ, то не существует и СПР. Лемма 3 доказана.

Перейдем к доказательству результатов о NP-сложности нахождения СПР для классов входов $\mathscr{I}(x_4)$ и $\mathscr{I}(x_5)$.

Теорема 7. *Задача* СПР *на классе входов* $\mathscr{I}(x_4)$, *где* $x_4 = (1, 2, 2, \infty)$, *разрешима за время* $O(n^2)$, *где* n *— число вершин графа* G.

Доказательство. Для решения задачи СПР на классе входов $I \in \mathscr{I}(x_4)$ достаточно применить алгоритм $\mathscr{A}^4$, состоящий из двух этапов.

• На первом этапе с помощью алгоритма $\mathscr{A}_\mu(I)$ решается задача СПМ.

• Если решение задачи СПМ положительно, то на втором этапе по найденной СПМ μ с помощью алгоритма $\mathscr{A}^4_{\mu,\rho}(I)$ строится СПР ρ.

Ясно, что если решение задачи СПР существует, то с помощью алгоритма $\mathscr{A}^4$ оно будет найдено. Если же СПР для входа I не существует, то из леммы 2 следует, что для I не существует и СПМ. Поскольку алгоритм $\mathscr{A}_\mu(I)$ решает задачу СПМ, то результатом его работы должно быть заключение: «СПМ для заданного входа не существует». Ясно также, что суммарная временная сложность алгоритмов $\mathscr{A}_\mu(I)$ и $\mathscr{A}^4_{\mu,\rho}(I)$ не превосходит $O(n^2)$, поскольку $u = O(n)$ для $I \in \mathscr{I}(x_4)$. Теорема 7 доказана.

Теорема 8. *Задача* СПР *на классе входов* $\mathscr{I}(x_5)$, *где* $x_5 = (1, 3, 2, 3)$, *разрешима за время* $O(n^2)$.

Доказательство теоремы 8 аналогично доказательству теоремы 7 (с точностью до замены алгоритма $\mathscr{A}^4_{\mu,\rho}(I)$ на алгоритм $\mathscr{A}^5_{\mu,\rho}(I)$ и леммы 2 на лемму 3).

4. Заключительные замечания и открытые вопросы

Хотя в разд. 2 было показано, что предложенная нами сложностная классификация классов входов задачи СПР является полной, это не исключает возможности классификации задачи по другим наборам параметров. Например, в определяемых нами классах $\mathscr{I}(x)$ задавались ограничения сверху на параметры $\Delta(G)$, $A_{\max}(G)$, $f(A)$ и NP-полнота достигалась при сравнительно малых значениях этих параметров. С другой стороны, если граф G близок к полному (и следовательно, степени вершин и $\Delta(G)$ очень велики), то связность области раскраски каждого цвета перестает быть ограничивающим фактором и проверка существования СПР становится полиномиально осуществимой. Таким образом, возможно существование сложностной классификации задачи СПР, основанной на ограничениях сверху параметра $\Delta(\overline{G})$, где $\overline{G}$ — дополнение графа G. Представляет интерес нахождение такой альтернативной классификации.

Другой открытой проблемой является определение сложности вычисления параметра $\alpha(G)$ для произвольного графа G (см. введение), выяснение зависимости этой сложности от максимальной степени графа G и значений его других параметров.

Авторы выражают благодарность В. Г. Визингу за постановку проблемы и интерес, проявленный к данным исследованиям. Кроме того, авторы благодарны анонимному рецензенту, предложившему ввести понятие допустимой маркировки, что позволило упростить доказательство теорем 7 и 8.

ЛИТЕРАТУРА

1. **Асратян А. С., Камалян Р. Р.** Интервальная раскраска ребер мультиграфа // Прикладная математика. Ереван: Изд-во Ереван. гос. ун-та, 1987. № 5. С. 25–34.
2. **Визинг В. Г.** О связной раскраске графов в предписанные цвета // Дискрет. анализ и исслед. операций. Сер. 1. 1999. Т. 6, № 4. С. 36–43.
3. **Гимади Э. Х.** О задаче размещения на сети с центрально-связными областями обслуживания // Управляемые системы: Сб. науч. тр. Новосибирск: Ин-т математики СО АН СССР, 1984. Вып. 25. С. 38–47.
4. **Севастьянов С. В.** Об интервальной раскрашиваемости ребер двудольного графа // Методы дискретного анализа в решении экстремальных задач: Сб. науч. тр. Новосибирск: Ин-т математики СО АН СССР. 1990. Вып. 50. С. 61–72.

5. **Ageev A. A.** A criterion of polynomial-time solvability for the network location problem // Integer programming and optimization. Carnegie Mellon Univ.: Campus Printing, 1992. P. 237–245.

6. **Hanson D., Loten C. O. M., Toft B.** On interval colourings of bi-regular graphs // ARS Combinatoria. 1998. V. 50. P. 23–32.

7. **Hopcroft J. E., Karp R. M.** An $n^{5/2}$ algorithm for maximum matching in bipartite graphs // SIAM J. Comput. 1973. V. 2, N 2. P. 225–231.

Адрес авторов:

Институт математики
им. С. Л. Соболева СО РАН,
пр. Академика Коптюга, 4,
630090 Новосибирск, Россия.
E-mail: seva@math.nsc.ru

Статья поступила
1 ноября 1999 г.